UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/595,776 | 06/16/2000 | Enric Musoll | P3810 | 1143 |

24739 7590 11/10/2003

CENTRAL COAST PATENT AGENCY
PO BOX 187
AROMAS, CA 95004

| EXAMINER | |
|---|---|
| HUISMAN, DAVID J | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2183 | 4 |

DATE MAILED: 11/10/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

| Office Action Summary | Application No. 09/595,776 | Applicant(s) MUSOLL ET AL. | |
|---|---|---|---|
| | Examiner David J. Huisman | Art Unit 2183 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *20 August 2003*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-13* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-13* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 June 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. §§ 119 and 120**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

13)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

a) ☐ The translation of the foreign language provisional application has been received.

14)☒ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

**Attachment(s)**

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____.
4) ☐ Interview Summary (PTO-413) Paper No(s). _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

## DETAILED ACTION

1. Claims 1-13 have been examined.

### *Papers Submitted*

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: #3. Amendment "A" as received on 8/20/2003.

### *Claim Objections*

3. Claim 6 is objected to because of the following informalities: In the last line of claim 6, replace "where" with --which--. Appropriate correction is required.

### *Maintained Rejections*

4. Applicant has failed to overcome the rejections set forth in the previous Office Action (mailed on 5/20/2003), which are hereby maintained by the examiner and copied below for applicant's convenience.

### *Maintained Claim Rejections - 35 USC § 103*

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-3 and 5-8, 10-11, and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Parady, U.S. Patent No. 5,933,627, in view of McFarling et al., U.S. Patent No. 5,758,142 (herein referred to as McFarling).

7. Referring to claim 1, Parady has taught in a multi-streaming processor having a data cache, a system for fetching instructions from individual ones of the multiple streams to a pipeline, comprising:

a) a fetch algorithm for selecting from which stream to fetch instructions. See the abstract.

b) Parady has not taught a hit/miss predictor for forecasting whether instructions will hit or miss the data cache wherein the prediction by the hit/miss predictor is used by the fetch algorithm in determining from which stream to fetch. However, McFarling has taught a hit/miss predictor that is used to predict, for load instructions, whether a cache hit or miss will occur. And, if a cache miss is predicted to occur, then instructions independent of the load are scheduled ahead of the load-dependent instructions. See column 3, lines 7-18. A person of ordinary skill in the art would have recognized that this prediction scheme would be useful in a multi-streaming (multithreaded) environment because a thread is a sequence (stream) of instructions that is independent from other threads (i.e. other sequences of instructions), as is known in the art. By implementing such a prediction scheme into the system of Parady, thread switches can occur sooner, thereby maximizing efficiency through execution of load-independent instructions. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the thread-switching system of Parady to include a hit/miss predictor as taught by McFarling, in order to switch threads before the load reaches the execution stage, thereby preventing a drop-off in throughput.

8. Referring to claim 2, Parady in view of McFarling has taught a system as described in claim 1. Furthermore, McFarling has taught that a hit prediction precipitates no change in the fetching process, i.e., the instructions dependent on the load will not be preempted by instructions independent of the load. See column 3, lines 7-18, and column 8, lines 28-49. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 1, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's system. In the combined Parady/McFarling system, if a hit prediction occurs, then instructions from the same thread will be fetched and executed (these would be instructions from the same stream).

9. Referring to claim 3, Parady in view of McFarling has taught a system as described in claim 1. Furthermore, McFarling has taught that a miss prediction results in switching fetching to a different stream, i.e., the instructions dependent on the load will be preempted by instructions independent of the load. See column 3, lines 7-18, and column 8, lines 28-49. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 1, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's system. In the combined Parady/McFarling system, if a miss prediction occurs, then instructions from a different thread will be fetched and executed (these would be instructions from a load-independent stream).

10. Referring to claim 4, Parady in view of McFarling has taught a system as described in claim 1. Furthermore, McFarling has taught the hit/miss predictor determines a hit probability, and the probability is used by the fetch algorithm in determining from where to fetch next instructions. See column 8, line 50, to column 9, line 33. Note that a 2-bit saturating counter

(doesn't wrap-around) exists for each load instruction within the program. This counter is incremented if the load hits the cache and decremented if the load misses the cache. The most significant bit is then used to make the prediction (i.e. if the counter equals 10 or 11, a hit is predicted, otherwise a miss is predicted). Therefore, it can be seen that if the load always hits, the counter will saturate at a binary value of 11, which would always result in a hit prediction. This predictor determines a hit probability in that each possible value of the counter represents a different probability based on past experience. The higher the counter value, the higher the hit probability. For instance, a counter value of 11 signifies a load instruction that has hit the cache recently. Therefore, it will be predicted with a high hit probability. On the other hand, a counter value of 00 signifies a load instruction that has missed the cache recently. Therefore, it will be predicted with no hit probability, i.e., it will be predicted to miss. Generally, these extreme values (00 and 11) are considered to be strongly taken and strongly not-taken prediction values, respectively. Counter values of 01 and 10 can be considered not-taken and taken prediction values, respectively, but the probability is less that it will miss and less that it will hit, respectively. As a result, it can be seen that as the counter value increases, so does the hit probability, thereby indicating that a load with a counter value of 11 is much more likely to hit the cache than a load with a counter value of 00, according to past encounters. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 1, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's system. Note from above that the prediction directly results in what instruction are executed next. Therefore, in the combined Parady/McFarling system, higher probability hit predictions

will result in fetching from a different stream. Likewise, low probability hit predictions (miss predictions) will result in fetching from the same stream.

11. Referring to claim 5, Parady in view of McFarling has taught a system as described in claim 1. Furthermore, McFarling has taught that the forecast of the hit/miss predictor is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units. See column 3, lines 7-18, and column 8, lines 28-49, and note that when a miss occurs, the dispatcher will dispatch instructions based on the prediction. For example, if a hit prediction occurs, the dispatcher will continue dispatching instructions that may be dependent on the load. However, if a miss prediction occurs, the dispatcher will dispatch instructions that are independent of the load so that they do not have to wait for the result. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 1, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's system. In the combined Parady/McFarling system the dispatcher would work in the same fashion, i.e., if a hit prediction occurs, instructions from the same stream which may be dependent on the load can continue to be dispatched, while if a miss prediction occurs, then instructions from a different thread (independent of the load) will be dispatched.

12. Referring to claim 6, Parady has taught a multi-streaming processor comprising:

a) a data cache. See Fig.1, component 56, and Fig.2, component 82.

b) a fetch algorithm for selecting from which stream to fetch instructions. See the abstract.

c) Parady has not taught a hit/miss predictor for predicting whether instructions will hit or miss the cache wherein a prediction by the hit/miss predictor is used by the fetch algorithm in determining from which stream to fetch. However, McFarling has taught a hit/miss predictor

that is used to predict, for load instructions, whether a cache hit or miss will occur. And, if a cache miss is predicted to occur, then instructions independent of the load are scheduled ahead of the load-dependent instructions. See column 3, lines 7-18. A person of ordinary skill in the art would have recognized that this prediction scheme would be useful in a multi-streaming (multithreaded) environment because a thread is a sequence (stream) of instructions that is independent from other threads (i.e. other sequences of instructions), as is known in the art. By implementing such a prediction scheme into the system of Parady, thread switches can occur sooner, thereby maximizing efficiency through execution of load-independent instructions. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the thread-switching system of Parady to include a hit/miss predictor as taught by McFarling, in order to switch threads before the load reaches the execution stage, thereby preventing a drop-off in throughput.

13. Referring to claim 7, Parady in view of McFarling has taught a processor as described in claim 6. Furthermore, McFarling has taught that a hit prediction precipitates no change in the fetching process, i.e., the instructions dependent on the load will not be preempted by instructions independent of the load. See column 3, lines 7-18, and column 8, lines 28-49. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 6, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's processor. In the combined Parady/McFarling system, if a hit prediction occurs, then instructions from the same thread will be fetched and executed (these would be instructions from the same stream).

14. Referring to claim 8, Parady in view of McFarling has taught a processor as described in claim 6. Furthermore, McFarling has taught that a miss prediction results in switching fetching to a different stream, i.e., the instructions dependent on the load will be preempted by instructions independent of the load. See column 3, lines 7-18, and column 8, lines 28-49. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 6, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's processor. In the combined Parady/McFarling system, if a miss prediction occurs, then instructions from a different thread will be fetched and executed (these would be instructions from a load-independent stream).

15. Referring to claim 9, Parady in view of McFarling has taught a processor as described in claim 6. Furthermore, McFarling has taught the hit/miss predictor determines a hit probability, and the probability is used by the fetch algorithm in determining from where to fetch next instructions. See column 8, line 50, to column 9, line 33. Note that a 2-bit saturating counter (doesn't wrap-around) exists for each load instruction within the program. This counter is incremented if the load hits the cache and decremented if the load misses the cache. The most significant bit is then used to make the prediction (i.e. if the counter equals 10 or 11, a hit is predicted, otherwise a miss is predicted). Therefore, it can be seen that if the load always hits, the counter will saturate at a binary value of 11, which would always result in a hit prediction. This predictor determines a hit probability in that each possible value of the counter represents a different probability based on past experience. The higher the counter value, the higher the hit probability. For instance, a counter value of 11 signifies a load instruction that has hit the cache recently. Therefore, it will be predicted with a high hit probability. On the other hand, a counter

value of 00 signifies a load instruction that has missed the cache recently. Therefore, it will be predicted with no hit probability, i.e., it will be predicted to miss. Generally, these extreme values (00 and 11) are considered to be strongly taken and strongly not-taken prediction values, respectively. Counter values of 01 and 10 can be considered not-taken and taken prediction values, respectively, but the probability is less that it will miss and less that it will hit, respectively. As a result, it can be seen that as the counter value increases, so does the hit probability, thereby indicating that a load with a counter value of 11 is much more likely to hit the cache than a load with a counter value of 00, according to past encounters. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 6, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's processor. Note from above that the prediction directly results in what instruction are executed next. Therefore, in the combined Parady/McFarling system, higher probability hit predictions will result in fetching from a different stream. Likewise, low probability hit predictions (miss predictions) will result in fetching from the same stream.

16. Referring to claim 10, Parady in view of McFarling has taught a processor as described in claim 6. Furthermore, McFarling has taught that the forecast of the hit/miss predictor is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units. See column 3, lines 7-18, and column 8, lines 28-49, and note that when a miss occurs, the dispatcher will dispatch instructions based on the prediction. For example, if a hit prediction occurs, the dispatcher will continue dispatching instructions that may be dependent on the load. However, if a miss prediction occurs, the dispatcher will dispatch instructions that are independent of the load so that they do not have to wait for the result. This feature is part of

McFarling's predictor, which as discussed in the rejection of claim 6, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's processor. In the combined Parady/McFarling system the dispatcher would work in the same fashion, i.e., if a hit prediction occurs, instructions from the same stream which may be dependent on the load can continue to be dispatched, while if a miss prediction occurs, then instructions from a different thread (independent of the load) will be dispatched.

17. Referring to claim 11, Parady has taught in a multi-streaming processor having a data cache (see Fig.1, component 56, and Fig.2, component 82), a method for fetching instructions from individual ones of multiple streams as instruction sources to a pipeline (see the abstract). Parady has not taught making a hit/miss prediction by a predictor as to whether instructions previously fetched will hit or miss the data cache, and if the prediction is a miss, altering the source of the fetch. However, McFarling has taught a hit/miss predictor that is used to predict, for load instructions, whether a cache hit or miss will occur. And, if a cache miss is predicted to occur, then instructions independent of the load are scheduled ahead of the load-dependent instructions. See column 3, lines 7-18. A person of ordinary skill in the art would have recognized that this prediction scheme would be useful in a multi-streaming (multithreaded) environment because a thread is a sequence (stream) of instructions that is independent from other threads (i.e. other sequences of instructions), as is known in the art. By implementing such a prediction scheme into the system of Parady, thread switches can occur sooner, thereby maximizing efficiency through execution of load-independent instructions. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the thread-switching system of Parady to include a hit/miss predictor as taught by McFarling, in

order to switch threads before the load reaches the execution stage, thereby preventing a drop-off in throughput.

18. Referring to claim 12, Parady in view of McFarling has taught a method as described in claim 11. Furthermore, McFarling has taught the hit/miss predictor determines a hit probability, and the probability is used by the fetch algorithm in determining from where to fetch next instructions. See column 8, line 50, to column 9, line 33. Note that a 2-bit saturating counter (doesn't wrap-around) exists for each load instruction within the program. This counter is incremented if the load hits the cache and decremented if the load misses the cache. The most significant bit is then used to make the prediction (i.e. if the counter equals 10 or 11, a hit is predicted, otherwise a miss is predicted). Therefore, it can be seen that if the load always hits, the counter will saturate at a binary value of 11, which would always result in a hit prediction. This predictor determines a hit probability in that each possible value of the counter represents a different probability based on past experience. The higher the counter value, the higher the hit probability. For instance, a counter value of 11 signifies a load instruction that has hit the cache recently. Therefore, it will be predicted with a high hit probability. On the other hand, a counter value of 00 signifies a load instruction that has missed the cache recently. Therefore, it will be predicted with no hit probability, i.e., it will be predicted to miss. Generally, these extreme values (00 and 11) are considered to be strongly taken and strongly not-taken prediction values, respectively. Counter values of 01 and 10 can be considered not-taken and taken prediction values, respectively, but the probability is less that it will miss and less that it will hit, respectively. As a result, it can be seen that as the counter value increases, so does the hit probability, thereby indicating that a load with a counter value of 11 is much more likely to hit

the cache than a load with a counter value of 00, according to past encounters. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 11, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's method. Note from above that the prediction directly results in what instruction are executed next. Therefore, in the combined Parady/McFarling system, higher probability hit predictions will result in fetching from a different stream. Likewise, low probability hit predictions (miss predictions) will result in fetching from the same stream.

19. Referring to claim 13, Parady in view of McFarling has taught a method as described in claim 11. Furthermore, McFarling has taught that the forecast of the hit/miss predictor is also used by a dispatch algorithm in selecting instructions from the pipeline to dispatch to functional units. See column 3, lines 7-18, and column 8, lines 28-49, and note that when a miss occurs, the dispatcher will dispatch instructions based on the prediction. For example, if a hit prediction occurs, the dispatcher will continue dispatching instructions that may be dependent on the load. However, if a miss prediction occurs, the dispatcher will dispatch instructions that are independent of the load so that they do not have to wait for the result. This feature is part of McFarling's predictor, which as discussed in the rejection of claim 1, would have been obvious to one of ordinary skill in the art at the time of the invention to implement in Parady's method. In the combined Parady/McFarling system the dispatcher would work in the same fashion, i.e., if a hit prediction occurs, instructions from the same stream which may be dependent on the load can continue to be dispatched, while if a miss prediction occurs, then instructions from a different thread (independent of the load) will be dispatched.

***Response to Arguments***

20. Applicant's arguments filed on August 20, 2003, have been fully considered but they are not persuasive.

21. In the remarks, Applicant argues the novelty/rejection of claims 1, 6, and 11 on pages 8-9 of the remarks, in substance that:

> "...the examiner does not appreciate the clear distinction between streams and threads in a multi-streaming processor," and as a result, "the examiner's interpretation that it would have been obvious to apply the teaching [McFarling's prediction scheme] for use in a multithreaded processor is clearly incorrect."

22. These arguments, and other arguments related to the above arguments, are not found persuasive for the following reasons:

a) The examiner asserts that the applicant may be reading the claims too broadly. The examiner has attached extrinsic evidence providing a dictionary definition of the word "stream" in which it is defined as "...a flow of data from a source (sender) to a single sink (receiver)." From looking at Fig.3, Parady has clearly taught multiple streams in that multiple instruction buffers exist (102-108) which provide instructions corresponding to different threads. Each instruction buffer is a "sender" of data (instructions) to a "receiver" (dispatch and execution units). As a result, Parady maps to applicant's claim 1, for instance, in that Parady has taught a multi-streaming processor, i.e., there are 4 possible instruction streams (one from each thread's instruction buffer flowing to the dispatch unit). And, instructions are fetched from individual ones of the multiple streams. See column 3, lines 40-43. The examiner then used McFarling to show a teaching of a hit/miss predictor that is used to predict, for load instructions, whether a cache hit or miss will occur, and, if a cache miss is predicted to occur, then instructions independent of the load are scheduled ahead of the load-dependent instructions. See column 3, lines 7-18, of McFarling.

This is relevant to Parady because a given thread (which also corresponds to a stream in Parady) is generally independent of other threads. Consequently, much like scheduling load-independent instructions ahead of instructions dependent on a load that is predicted to miss the cache (in McFarling), one of ordinary skill in the art would have recognized that thread-independent instructions (i.e., instructions from a second thread) could be scheduled ahead of first thread's instructions that are dependent on a load in the first thread that is predicted to miss the cache. More specifically, in Parady in view of McFarling, if instructions were being fetched from a first stream (thread 0's instruction buffer) and a load instruction within thread 0's stream of instructions is predicted to miss the cache, then according to McFarling, instructions from a second stream would be dispatched, say from thread 1's instruction buffer, in order to prevent stalling. This is because thread 1's instructions would be independent of thread 0 and they would not be dependent on a load within thread 0 that would possibly miss the cache. By implementing such a prediction scheme into the system of Parady, thread switches would occur sooner, thereby maximizing efficiency through execution of load-independent instructions. Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify the thread-switching system of Parady to include a hit/miss predictor as taught by McFarling, in order to switch threads before the load reaches the execution stage, thereby preventing a drop-off in throughput.

## *Conclusion*

23. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (703) 305-7811. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is (703) 746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

DJH
David J. Huisman
November 6, 2003

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100